

Programação Visual

Trabalho de Laboratório nº 10

Objetivo	Angular – Introdução à utilização de Angular em projetos ASP.NET Core MVC.
Programa	Pretende-se criar uma aplicação cliente dos serviços da aplicação ESTeSoccer usando angular.
Regras	Implementar o código necessário e testar no fim de cada nível. Use as convenções de codificação adotadas para as linguagens usadas.
Descrição	
Nível 1	<ul style="list-style-type: none">• Descarregue e descompacte o ficheiro “Lab 10 – Materiais” fornecido com este enunciado. Depois de descompactar, abra a solução que lá está. Nessa solução irá encontrar 2 projetos:<ol style="list-style-type: none">1. ESTeSoccer – É uma aplicação MVC que disponibiliza uma API com ações CRUD para as ligas de futebol guardadas na base de dados..2. ESTeSoccerClient – É uma aplicação de consola que gere os dados das ligas de futebol usando os serviços disponibilizados pelo controlador LeaguesAPIController da aplicação anterior.• Corra os comandos Add-Migration e Update-Database.• A aplicação ESTeSoccer usa a base de dados criada e quando for executada pela primeira vez irá colocar alguns dados iniciais. Sendo assim, execute esta aplicação e verifique que está a funcionar, confirme que a utilização dos serviços de consulta das ligas de futebol do controlador LeaguesController retorna a informação das 2 ligas de futebol que lá foram colocadas inicialmente.• Sem parar a execução anterior, arranque o projeto ESTeSoccerClient da mesma forma e verifique que também está a funcionar. Deverá ver na listagem das ligas, as 2 ligas adicionadas antes.• Dentro da solução, crie uma aplicação Frontend usando a framework Angular. Crie um novo projeto com o nome ESTeSoccerAngularClient do tipo ASP.NET Core with Angular.• Teste a aplicação criada, deverá ver uma página “Hello World”. <p>Notas:</p> <ul style="list-style-type: none">• Atenção que para esta aplicação funcionar poderá ter que instalar o NodeJs (Se for necessário, encontra este servidor em https://nodejs.org/en/).• Pode demorar algum tempo a iniciar.• Poderá aparecer algum tipo de configuração relativamente à partilha de dados anónimos.
Nível 2	<ul style="list-style-type: none">• Pretende-se agora criar uma aplicação cliente semelhante à aplicação de consola existente mas utilizando uma interface desenvolvida em Angular. Comece então por alterar o componente Angular Home (“ClientApp/src/app/home”) para que passe a mostrar o texto “ESTeSoccer” no título. Na barra de menu em cima também deverá aparecer “ESTeSoccer” (componente nav-menu).

Programação Visual

Trabalho de Laboratório nº 10

- Crie um novo componente Angular com o nome “**About**” e adicione-o ao projeto.
- Abra uma linha de comandos, no diretório do Angular (ClientApp) e corra o seguinte comando, que irá gerar um novo component com os ficheiros necessários:
ng generate component about --module=app

- Abra o novo componente e troque adicione no html o seu nome.
- Altere o component nav-menu e adicione uma nova opção ao menu para aceder ao novo componente.
- Altere o ficheiro app.module.ts e acrescente no RouterModule um novo valor para o novo componente.

Nível 3

- Precisamos agora de um componente mostre as ligas de futebol, usando o serviço correspondente da aplicação **ESTeSoccer**. Uma vez que pretendemos chamar um servidor diferente de onde estamos a correr o angular, iremos necessitar de um proxy para redirectionar os nossos pedidos para o servidor correto. Para tal, vamos fazer algumas configurações na aplicação Angular:
- No ficheiro **proxy.conf.js** adicione no array PROXY_CONFIG o seguinte objecto:

```
{
  context: [
    "/api ",
  ],
  target: "https://localhost:7040",
  secure: false
}
```

Desta forma, sempre que um URL comece com **/api** será utilizado o URL <https://localhost:7040/api> seguido da restante informação.

- Agora já pode utilizar o serviço que devolve as ligas de futebol existentes. Para isso, defina um novo componente **leagues**, e adicione uma nova opção ao menu (não se esqueça de alterar o RouterModule).

Sugestão: baseie-se no código do componente **FetchDataComponent**.

Notas: Dentro do código do componente Leagues, não necessita de receber o BASE_URL, basta usar apenas `get<League[]>('api/leaguesapi')`.

Na interface **League** (que irá criar) não necessita de incluir a propriedade para as equipas.

Nível 4

- Crie um componente que permita criar uma nova liga (receba o nome e país).

Consulte a documentação de como criar um formulario e de como enviar dados para um servidor:

<https://angular.io/start/start-forms#define-the-checkout-form-model>

<https://angular.io/guide/http#sending-data-to-a-server>

Nível 5

- Altere ao componente das ligas, e adicione um metodo que permita apagar uma liga. Acrescente na listagem um botao em cada linha para efetuar esta ação.

Programação Visual

Trabalho de Laboratório nº 10

Desafio

- Adicione à lista das ligas 2 botões, um para ver os detalhes, que abre uma nova página com os detalhes da liga selecionada, e outro para editar uma liga, que abre uma nova página com os dados da liga e que os permite alterar.

Notas

Para os identificadores do C# siga as convenções adotadas, nomeadamente:

- A notação camelCase para o nome das variáveis locais e identificadores privados.
- A notação PascalCase para os nomes públicos dos métodos e classes
- Não utilize o símbolo de qualquer forma '_' nos identificadores
- Não use abreviaturas